

PAPER • OPEN ACCESS

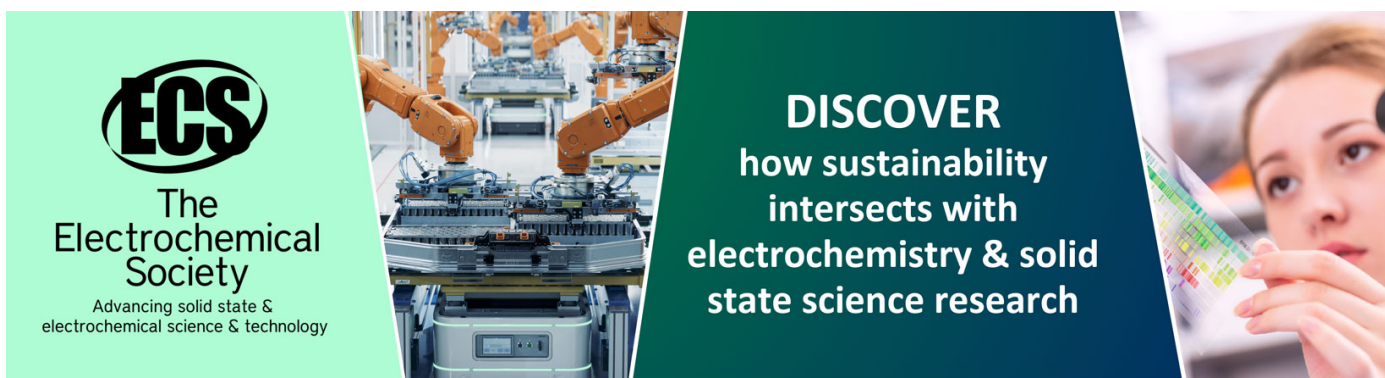
Pitch control of wind turbine based on deep neural network

To cite this article: Wei Jie *et al* 2020 *IOP Conf. Ser.: Earth Environ. Sci.* **619** 012034

View the [article online](#) for updates and enhancements.

You may also like

- [Comparison of Current Mirrors Designed with TFET or FinFET Devices for Different Dimensions and Temperatures](#)
Marcio Dalla Valle Martino, Joao Antonio Martino, Paula Ghedini Der Agopian et al.
- [Optimal Control of Li-Ion Battery Energy Storage System for Frequency Regulation](#)
Jihun LIM, Jin Hyeok Choi, Geon-pyo Lim et al.
- [Profile-Decomposing Output of Multi-Channel Odor Sensor Array](#)
Xiaofan Zheng, Yoichi Tomiura, Kenshi Hayashi et al.



ECS
The
Electrochemical
Society
Advancing solid state &
electrochemical science & technology

DISCOVER
how sustainability
intersects with
electrochemistry & solid
state science research

Pitch control of wind turbine based on deep neural network

Wei Jie, Chu Jingchun, Yuan Lin, Wang Wenliang, Dong Jian

Guodian United Power Technology Co., LTD. , Yard 16, West Fourth Ring Road,
Haidian District, Beijing

12032845@chnenergy.com.cn

Abstract. This paper analyzed the input and output data of wind farm based on deep neural network, developed intelligent model, and realized the predictive modeling of important parameter variables and control of wind turbine. By establishing the Deep Extreme Learning Machine(DELM), the higher-order nonlinear model is simplified. In this structure, unsupervised hierarchical ELM is conducted for feature extraction, and the features of the lower layer are transferred to the higher layer through layer by layer coding to form a relatively complete feature representation. Finally, the Extreme Learning Machine (ELM) is used to complete the mapping of feature representation to target output to minimize the loss of information in the transmission process. The target output is used as reference data for Pitch control of wind turbine, which is proposed by using a radial basis function (REF) neural network. Simulation results from GH-Bladed show that proposed control algorithm can mitigate the loads effectively. The algorithm provides a practical reference for the design of wind turbine controller.

1. introduction

Wind turbine is a complex nonlinear system. The more factors are considered, the higher the order of the system will be. Moreover, the dynamic responses of various factors vary greatly, which will cause the wind turbine to become a complex system. All of this is bad for the design of the controller, which makes it very complex and sensitive to parameters [1].

At present, nonlinear adaptive control method and modeling have become the research hotspot in the field of wind turbine control. Compared with the traditional control strategy, it does not require accurate mathematical model of the controlled object and is more suitable for the actual demand [2-3]. Nonlinear adaptive control method can adjust the control parameters with self-learning ability, and improve the control accuracy by fully utilizing sensors [4].

Therefore, this paper hopes to realize the analysis of wind farm input and output data through the design of intelligent control algorithm, and the predictive modeling and control of wind turbines or important parameter variables. By establishing an optimal data-driven soft measurement model, the high-order nonlinear model is simplified. This paper used deep neural network as new intelligent modeling method, and completed the code development, and set up the simulation platform.

2. design

PI controller is widely used to control the pitch Angle of turbine blade due to its simplicity and stability. However, neural networks and evolutionary algorithms provide a new way of thinking for PI gain optimization [5-6]. In this design, PI controller based on RBF neural network is used as the pitch



controller of wind turbine. In order to obtain the optimal data set of training RBF neural network, PSO evolutionary algorithm is adopted.

In this design, wind speed value is predicted by Deep Extreme Learning Machine(DELM), which is used as the input of the RBF neural network, and the proportional and integral gain of PI controller is used as the output of RBF neural network. In order to obtain the optimal training data set, this design uses PSO algorithm to optimize the proportional gain and integral gain at a specific wind speed greater than the rated wind speed. By adjusting the blade Angle, the trained controller can minimize the difference between the measured and rated value of generator speed.

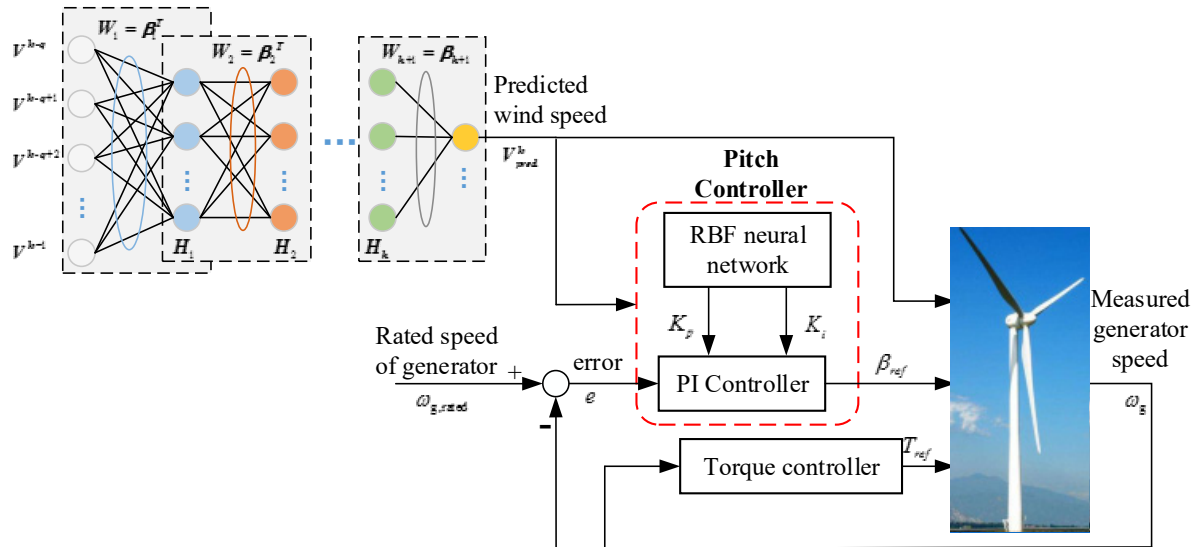


Fig. 1. Pitch control block diagram of wind turbine based on deep neural network.

3. Wind speed prediction based on DELM

3.1. Extreme Learning Machine

Different from traditional gradient-based algorithms, ELM's input weights and single-hidden layer biases are arbitrarily chosen without iterative adjust, and the only parameters to be learned in training are the output weights which can be calculated by solving a single linear system.

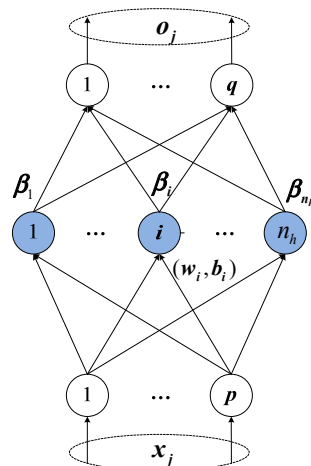


Fig. 2. The model of basic ELM.

Given N training samples, $\{X, T\} = \{x_j, t_j\}_{j=1}^N$, where x_j and t_j are the j -th input and target values respectively. To seek a regressor function from the input to the target [7-9], the standard Single Hidden Layer Feed-forward network can be mathematically modeled as:

$$o_j = \sum_{i=1}^{n_h} \beta_i g(w_i^T x_j + b_i) = t_j \quad (1)$$

where o_j is the output vector of the j -th training sample, w_i is the input weight vector connecting the input nodes to the i -th hidden node, b_i is the bias of the i -th hidden node, $g(\cdot)$ denotes hidden nodes nonlinear piecewise continuous activation functions.

The above N equations can be written compactly as:

$$H\beta = T \quad (2)$$

where the matrix T is target matrix,

$$H = \begin{bmatrix} g(w_1^T x_1 + b_1) & L & g(w_{n_h}^T x_1 + b_{n_h}) \\ M & L & M \\ g(w_1^T x_N + b_1) & L & g(w_{n_h}^T x_N + b_{n_h}) \end{bmatrix} \quad (3)$$

$$\beta = \begin{bmatrix} \beta_1^T \\ M \\ \beta_{n_h}^T \end{bmatrix}, T = \begin{bmatrix} t_1^T \\ M \\ t_N^T \end{bmatrix}. \quad (4)$$

The matrix H is the hidden layer output matrix, which can be randomly generated independent of the training data. $\beta = [\beta_1, \beta_2, \dots, \beta_{n_h}]^T$ is the output weight matrix between the hidden nodes and the output nodes. Thus, training SLFNs simply amounts to getting the solution of a linear system (2) of output weights β [10-12].

A simple representation of the solution of the equation (2) is given explicitly by Huang et al. [12] as

$$\hat{\beta} = H^\dagger T \quad (5)$$

where $H^\dagger$ is the Moore-Penrose generalized inverse of the hidden layer output matrix H .

To improve generalization performance and make the solution more robust, we can add a regularization term [13], as shown in the equation (6) and equation (7),

$$\hat{\beta} = \left(\frac{I}{\lambda} + H^T H \right)^{-1} H^T T, \quad (6)$$

$$\hat{\beta} = H^T \left(\frac{I}{\lambda} + H^T H \right)^{-1} T. \quad (7)$$

Thus, the ELM tends to reach the solutions straightforward without the issue of over-fitting. These two features make ELM more flexible and attractive than traditional gradient-based algorithms.

3.2. Deep Extreme Learning Machine (DELIM)

The Deep Extreme Learning Machine integrates the idea of Auto-Encoder in the algorithm, and codes the output by minimizing the reconstruction error, so that the output can infinitely approximate the original input [14-16]. This structure can capture relevant higher-level abstractions of the input.

Fig.3 illustrates the process of learning representation. The prediction modeling process of DELIM for the output variable Y_{t+s} at time $t+s$ takes the extracted training sample $\{\mathcal{S}_i\}_{i=1}^k$ as the input of DELIM

network. Here, we consider a fully connected multi-layer network with h hidden layers. Let $L = \{W_1, W_2, \dots, W_{h+1}\}$ denotes the parameters of the network that need to be learned.

In our paper, the Deep ELM is applied to learning the parameters L , which is designed by using the encoded outputs to approximate the original inputs by minimizing the reconstruction errors. The output weights β can be analytically determined by the equation (6) or (7) depending on the number of nodes in the hidden layer.

Each layer in the network can be decoupled as an independent ELM, and the target output T of each ELM can be equal to the input of the ELM [17-18]. In this way, you can get a low-dimensional representation of the input data, that is, the hidden output of ELM, which is the input of the next ELM. By means of the trained DELM model, the wind speed at the moment $t+s$ can be predicted.

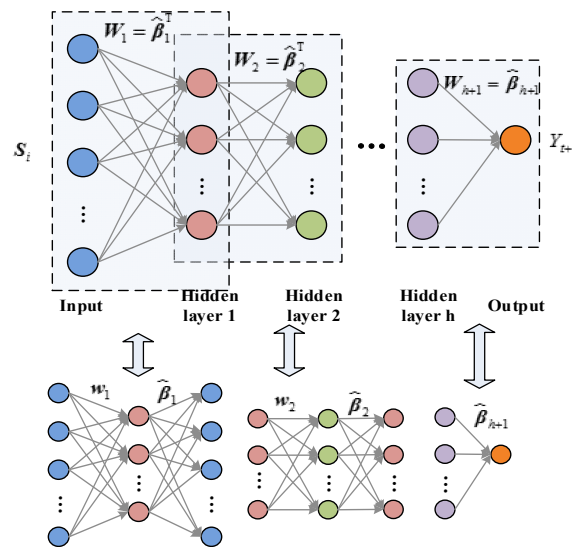


Fig. 3. Detailed illustration of the DELM representation learning.

4. Pitch control based on RBF

Artificial neural network and genetic algorithm are widely used in the design of the control system, while the pitch control of wind turbine uses the Radial Basis Function (RBF) neural network and Particle Swarm Optimization (PSO) evolutionary algorithm.

4.1. Radial basis function (REF) neutral network

Artificial Neural Networks (ANN) is a mathematical model which is inspired by Biological Neural Network and proposed to simulate the process of human brain information processing. It can realize features learning, classification and regression and other functions.

The mapping relationship between the input and output of neurons is:

$$y = g\left(\sum_{j=1}^n w_j x_j - b\right) \quad (8)$$

where x_j and w_j are the input signal and weight of the j -th neuron respectively. $g(\cdot)$ is the activation function, the common activation functions are $\text{traingd}(\cdot)$ 、 $\text{RBF}(\cdot)$ 、 $\text{sigmoid}(\cdot)$ etc.

And the neural network using $\text{RBF}(\cdot)$ as the activation function is RBF neural network. A typical RBF neural network is a three-layer structure: the input layer, the hidden layer with nonlinear RBF activation function and the linear output layer. Its structure is shown in fig.4.

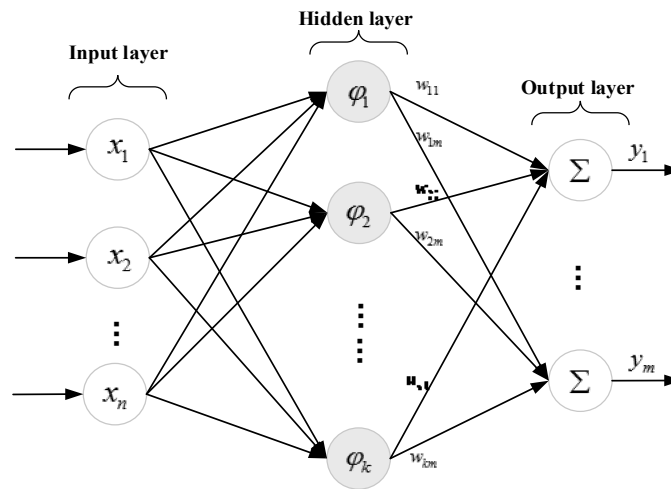


Fig. 4. RBF neural network structure diagram

4.2. PSO evolutionary algorithm

Particle swarm optimization (PSO) is a kind of evolutionary computing technology, which derives from the research on the predation behavior of birds. The basic idea of particle swarm optimization algorithm is to find the optimal solution through the cooperation and information sharing among individuals in the group.

PSO is initialized to a group of random particles (random solutions), and then iterated to find the optimal solution. In each iteration, the particle updates itself by tracking two "extreme values" (y_{ipbest}^k , y_{gbest}^k). After finding these two optimal values, the particle updates its velocity s_i^{k+1} and position y_i^{k+1} by the equation (9) and (10).

$$y_i^{k+1} = y_i^k + s_i^{k+1} \quad (9)$$

$$s_i^{k+1} = \omega s_i^k + c_1 r_1 (y_{ipbest}^k - y_i^k) + c_2 r_2 (y_{gbest}^k - y_i^k) \quad (10)$$

Where ω is inertia weight, c_1, c_2 are acceleration constants, and $r_1, r_2 \in (0, 1)$ is random number.

4.3. Pitch control process

A new pitch control method for wind turbines is proposed in this paper, as shown in the fig.1. In this method, based on the Extreme Learning Machine, the effective predictive value of wind speed is first obtained, which is detailed in section 3. Then, taking the predictive wind speed as an important input of the turbine control system, the optimal PI controller is designed to minimize the difference between the measured value and the rated value of the generator speed [19-20]. In order to achieve this goal, the PI controller should provide an appropriate pitch Angle reference value β_{ref} . The calculation formula is as follows:

$$\beta_{ref}(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau \quad (11)$$

Here, K_p and K_i respectively represent the proportion and integral gain of PI controller, and $e(t)$ can be calculated by the following formula:

$$e(t) = \omega_{g,rated} - \omega_g(t) \quad (12)$$

$\omega_{g,rated}$ and $\omega_g(t)$ represent the rated and measured values of the generator speed respectively. In this paper, RBF neural network is used to set the gain of PI controller. The predicted wind speed value

is used as the input of the neural network, and the proportional and integral gain of PI controller is used as the output of RBF neural network.

As shown in fig. 1, the reference value T_{ref} of generator torque is calculated by torque calculator in fig. 1. By adjusting the reference value of the pitch angle β_{ref} , the pitch controller in this design can provide an appropriate pitch angle β . By changing the pitch angle β and rotor speed ω_r , the generator speed ω_g can be finally changed. Therefore, if the controller is designed to perform well, the generator speed ω_g should be kept near the generator rated speed $\omega_{g,rated}$, in which case both the generator power P_g and the generator torque T_g should be kept near their rated values.

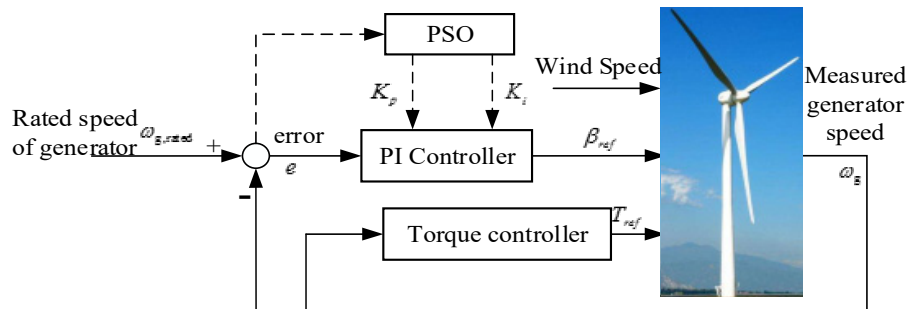


Fig. 5. Data set optimization process based on PSO evolutionary algorithm

In order to design the optimal performance of the PI pitch controller, the RBF neural network must be trained with the optimal training data set, so that the RBF neural network can provide the optimal PI gain. In order to obtain the optimal training data set, PSO evolutionary algorithm is used in this design. PSO algorithm is used to optimize the proportional gain and integral gain at a specific wind speed greater than the rated wind speed, as shown in the fig.5. The whole optimization process is completed by minimizing the cost function, namely the integral absolute error:

$$IAE = \int_0^{t_{sim}} |e(t)| dt \quad (13)$$

After initializing the population size and particle position and velocity, PSO calculates the PI controller gain of each particle and stores its IAE value. According to IAE value, the optimal position of each particle can be calculated, and the optimal particle can also be selected from all the population particles. The whole process iterates until the number of cycles reaches the maximum number of iterations. After the last cycle, the optimal particle is selected. In fact, the position information of the optimal particle contains the proportion and integral gain corresponding to the minimum IAE value.

PSO can only provide a pair of gains for constant wind speed, but we can obtain an optimal data set in this way. In this design, RBF neural network is selected to calculate the optimal PI gain for each wind speed based on the above optimal data set. When the network is trained, the constant wind speed in the optimal data set is taken as input and the corresponding PI gain is taken as output. After training, the RBF neural network can be used to calculate the optimal PI gain corresponding to any wind speed in the full load area with its general approximation characteristics.

5. Simulation results

In order to evaluate the performance of the proposed wind speed prediction model based on DELM, it is applied to turbulence wind prediction of 2MW wind turbine, as shown in fig. 6 in which 10 m/s average wind speed is used. This wind speed profile is obtained based on kaimal wind model. It can be seen from fig. 6 that the use of DELM for wind speed prediction can effectively track the real-time change of wind speed, which provides the possibility for the advance pitch control.

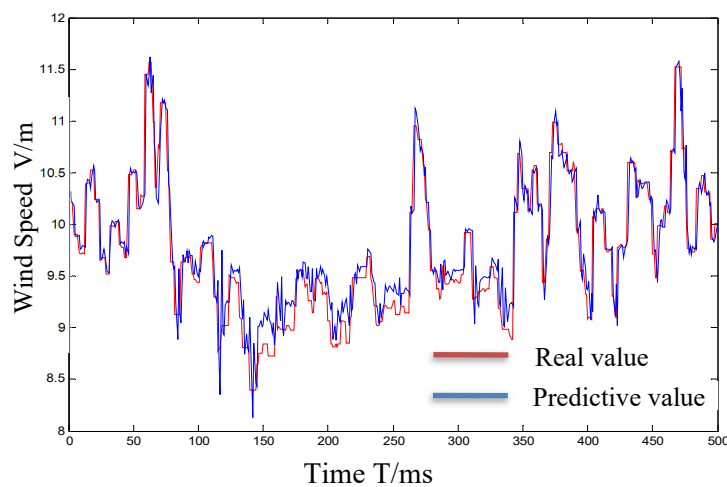


Fig. 6. Wind speed prediction results with average wind speed of 10m/s

Fig. 7 shows that amplitude of the ultimate load in the direction of the blade M_y , in which proposed controller based on RBF is less than the conventional PI pitch controller.

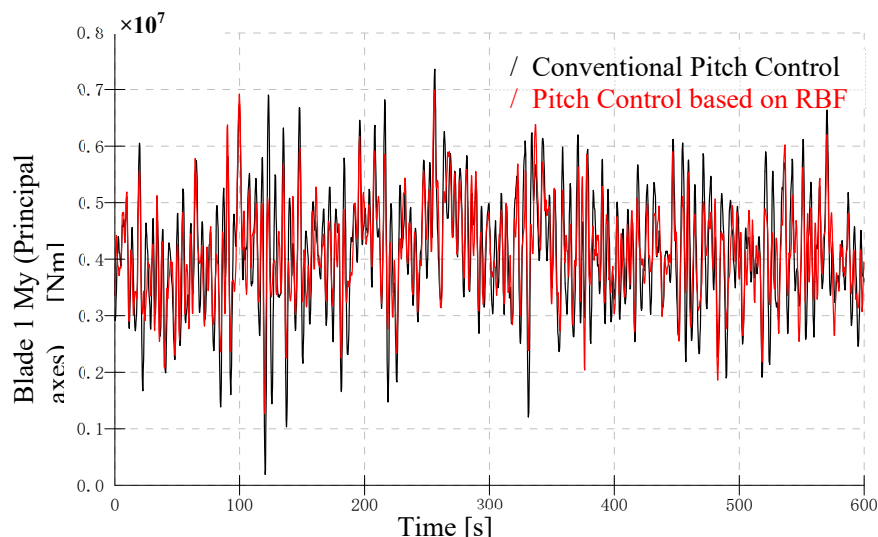


Fig. 7. Comparison of ultimate load in the direction of blade M_y

Therefore, according to the simulation results, compared with conventional PI controller, the proposed controller with RBF neural network has more effective performance in pitch angle control. In this way, the ultimate load of the blade can be effectively reduced and the service life of the wind turbine can be prolonged.

6. Conclusion

This paper analyzes the input and output data of wind farm through the design of intelligent control algorithm, develops intelligent model, and realizes the predictive modeling and control of important parameter variables of wind turbine. The new intelligent method of deep neural network can reduce the computational complexity of the whole model, reduce the consumption of computing resources, and realize the modeling and prediction of wind turbines or important parameter variables.

References

- [1] Xing G, Guo W. Method for collective pitch control of wind turbine generator system[J].

- Transactions of the Chinese society of agricultural engineering, 2008, 24(5):181-186.
- [2] Xing Z X, Zheng Q L, Yao X J, et al. PID control in adjustable-pitch wind turbine system based on BP neural network[J]. Journal of Shenyang University of Technology , 2006, 28(6):681-686.
 - [3] Xu L F, Xu D P, Gao F, et al. Compound pitch-control of wind turbine generator based on neural network[J]. Journal of North China Electric Power University , 2009, 36(1):28-34.
 - [4] Qin B, Zhou H, Du K, et al. Sliding Mode Control of Pitch Angle Based on RBF Neural-Network[J]. Transactions of China Electric technical Society , 2013, 28(5):37-41.
 - [5] He Y L, Liu J, Li J, et al. Variable-speed variable-pitch wind turbine control strategy optimization[J]. Power System Protection and Control, 2011, 39(12):55-60.
 - [6] Song X F, Liu J, Huang G. Variable Pitch Control of Wind Power Generation Based on RBF Neural Network Tuning PID Control[J]. Power system and clean energy, 2009, 25(4):49-53.
 - [7] Kang S L, Liu L, Liu C C, et al. Intrusion detection based on multiple layer extreme learning machine[J]. Journal of Computer Applications, 2015: 0-0.
 - [8] Chamara L L, Zhou H, Huang G B. Extreme learning machines—representational learning with ELMs for big data[J]. Intelligent Systems, IEEE, 2013, 28(6): 31-34.
 - [9] Yu W, Zhuang F, He Q, et al. Learning deep representations via extreme learning machines[J]. Neurocomputing, 2015, 149: 308-315.
 - [10] Zhu W, Miao J, Qing L, et al. Hierarchical extreme learning machine for unsupervised representation learning[C]//Neural Networks (IJCNN), 2015 International Joint Conference on. IEEE, 2015: 1-8.
 - [11] Uzair M, Shafait F, Ghanem B, et al. Representation learning with deep extreme learning machines for efficient image set classification[J]. arXiv preprint arXiv:1503.02445, 2015.
 - [12] Tang J, Deng C, Huang G B. Extreme learning machine for multilayer perceptron[J]. 2015.
 - [13] Cai Z, Tiwari R C. Application of a local linear autoregressive model to BOD time series[J]. Environmentrics, 2000, 11(3): 341-350.
 - [14] Gan L. Research and application of limit learning machine[D]. Xidian University , 2014.
 - [15] Li X D. The theory and algorithm of nuclear limit learning machine and its application in image processing[D]. Zhejiang University , 2014.
 - [16] Pinkus A. Approximation theory of the MLP model in neural networks[J]. Acta Numerica, 1999, 8: 143-195.
 - [17] Fu H. Semi-supervised classification based on limit learning machine[D].Xidian University, 2013.
 - [18] Huang G B, Zhu Q Y, Siew C K. Extreme learning machine: Theory and applications[J]. Neurocomputing, 2006, 70(1): 489-501.
 - [19] Poultangari I , Shahnazi R , Sheikhan M . RBF neural network based PI pitch controller for a class of 5-MW wind turbines using particle swarm optimization algorithm[J]. ISA Transactions, 2012, 51(5):641-648.
 - [20] Liu Q , Yue J . Pitch control of variable speed constant frequency wind turbines based on neural network controller[C]// Control Conference. IEEE, 2011.